

CLAIM AMENDMENTS

1 1. (Previously Presented) A method for encryption and decryption of electronic
2 messages based on an encryption protocol, the method comprising the
3 computer-implemented steps of:
4 receiving a first electronic message that is encrypted according to the encryption
5 protocol;
6 generating at least one part of a second electronic message, based on at least the first
7 electronic message, a modular operation that is based on two applications of
8 Montgomery's method, a first operand, a second operand, a modulus, and a
9 modular reduction using a negative multiplicative inverse of the modulus, and
10 wherein the step of generating the second electronic message includes the
11 computer-implemented steps of:
12 generating a first constant (R) based on the modulus (M) by selecting a second
13 constant (W) such that $W \geq 4M$ and determining the first constant (R)
14 according to the expression $R = W^2 \pmod{M}$;
15 in a first application of Montgomery's method, determining an intermediate
16 result based on at least Montgomery's method for the modular
17 operation, the first operand, and the first constant; and
18 in a second application of Montgomery's method, determining and storing in
19 memory a final result that comprises the at least one part of the second
20 electronic message, based on at least Montgomery's method for the
21 modular operation, the intermediate result, and the second operand.

1 2-6. (Cancelled)

1 7. (Previously Presented) The method of Claim 1, wherein the second constant (W) is
2 not a power of two.

1 8. (Cancelled)

1 9. (Currently Amended) The method of Claim 1, wherein the modular operation for
2 determining the intermediate result is a modular multiplication that is based on at
3 least the first operand, the first constant, a second constant, the modulus, and a the
4 negative multiplicative inverse of the modulus, and wherein the intermediate result is
5 determined by the computer-implemented steps of:
6 determining a modified first operand based on the first operand and the first constant;
7 determining a modular reduction of the modified first operand based on the modified
8 first operand, the negative multiplicative inverse of the modulus, and the
9 second constant; and
10 determining the intermediate result based on the modified first operand, the modular
11 reduction of the modified first operand, the modulus, and the second constant.

1 10. (Cancelled)

1 11. (Previously Presented) The method of Claim 1, wherein the modular operation for
2 determining the final result is a modular multiplication that is based on at least the
3 second operand, a second constant, the modulus, the negative multiplicative inverse
4 of the modulus, and the intermediate result, and wherein the final result is determined
5 by the computer-implemented steps of:
6 determining a modified second operand based on the second operand and the
7 intermediate result;
8 determining a modular reduction of the modified second operand based on the
9 modified second operand, the negative multiplicative inverse of the modulus,
10 and the second constant; and
11 determining the final result based on the modified second operand, the modular
12 reduction of the modified second operand, the modulus, and the second
13 constant.

1 12. (Cancelled)

1 13. (Previously Presented) The method of Claim 1, wherein the modular operation for
2 determining the intermediate result is a modular exponentiation that is based on at
3 least the first operand, the first constant, a second constant, the modulus, and the
4 negative multiplicative inverse of the modulus, and wherein the intermediate result is
5 determined by the computer-implemented steps of:
6 determining a modified first operand based on the first operand and the intermediate
7 result;
8 determining a modular reduction of the modified second operand based on the
9 modified second operand, the negative multiplicative inverse of the modulus,
10 and the second constant; and
11 determining the intermediate result based on the modified second operand, the
12 modular reduction of the modified second operand, the modulus, and the
13 second constant.

1 14. (Cancelled)

1 15. (Previously Presented) The method of Claim 1, wherein the modular operation for
2 determining the final result is a modular exponentiation that is based on at least the
3 second operand, a second constant, the modulus, the negative multiplicative inverse
4 of the modulus, and the intermediate result, wherein the second operand includes a
5 plurality of digits, and wherein the final result is determined by the
6 computer-implemented steps of:
7 specifying a previous final result as having a value of one and a previous intermediate
8 result as the intermediate result;
9 for each digit of the plurality of digits included in the second operand, performing the
10 computer-implemented steps of:
11 when each digit of the plurality of digits has the value of one, then performing
12 the computer-implemented steps of:
13 determining an updated final result based on the previous final result
14 and the previous intermediate result;

15 determining a modular reduction of the updated final result based on
16 the updated final result, the negative multiplicative inverse of
17 the modulus, and the second constant;
18 determining a revised updated final result based on the updated final
19 result, the modular reduction of the final result, the modulus,
20 and the second constant;
21 if all digits of the plurality of digits have not been evaluated,
22 specifying the revised updated final result as the previous final
23 result; and
24 if all digits of the plurality of digits have been evaluated, specifying the
25 revised updated final result as the final result;
26 determining an updated intermediate result based on the previous intermediate
27 result;
28 determining a modular reduction of the updated intermediate result based on
29 the updated intermediate result, the negative multiplicative inverse of
30 the modulus, and the second constant;
31 determining a revised updated intermediate result based on the updated
32 intermediate result, the modular reduction of the updated intermediate
33 result, the modulus, and the first constant; and
34 specifying the revised updated intermediate result as the previous intermediate
35 result.

1 16. (Cancelled)

1 17. (Original) The method of Claim 1, further comprising the computer-implemented
2 steps of:
3 generating a plurality of residual number system (RNS) representations, wherein the
4 plurality of RNS representations includes at least one RNS representation for
5 each of the first operand, the modulus, and the first constant;

6 wherein the step of determining the intermediate result includes the
7 computer-implemented step generating the intermediate result based on
8 Montgomery's method for the modular operation and the plurality of RNS
9 representations; and
10 wherein the step of determining the final result includes the computer-implemented
11 step of generating the final result based on Montgomery's method for the
12 modular operation and the plurality of RNS representations.

1 18. (Previously Presented) The method of Claim 17, wherein:
2 the plurality of RNS representations includes a first set of RNS representations in a
3 first RNS base and a second set of RNS representations in a second RNS base;
4 the first RNS base extends the second RNS base;
5 the first RNS base includes a first group of sixty-four residues and the second RNS
6 base includes a second group of sixty-four residues;
7 each residue in both the first group and second group is a seventeen-bit residue;
8 each residue in the first group is relatively prime with respect to all other residues in
9 the first group and each residue in the second group is relatively prime with
10 respect to all other residues in the second group;
11 each residue in both the first group and the second group are selected from a range
12 of 2^{16} to 2^{17} ; and
13 the method further comprises the computer-implemented steps of:
14 converting, in eight clock cycles, a first RNS representation of the first set of
15 RNS representations in the first RNS base to a second RNS
16 representation of the second set of RNS representations in the second
17 RNS base;
18 performing operations involving each residue in the first group in parallel; and
19 performing operations involving each residue in the second group in parallel.

1 19-26. (Cancelled)

1 27. (Previously Presented) The method of Claim 1, wherein the modular operation is a
2 modular multiplication and the step of generating the second electronic message
3 further includes the computer-implemented step of:
4 while determining the intermediate result and determining the final result, storing
5 results of intermediate computations in a first register file and a second
6 register file, wherein the first register file includes a first set of sixty-four
7 seventeen-bit registers and the second register file includes a second set of
8 sixty-four seventeen-bit registers.

1 28. (Cancelled)

1 29. (Previously Presented) The method of Claim 1, wherein the modular operation is a
2 modular exponentiation and the step of generating the second electronic message
3 further includes the computer-implemented step of:
4 while determining the intermediate result and determining the final result, storing
5 results of intermediate computations, in a first register file a second register
6 file, a third register file, and a fourth register file; and
7 wherein the first register file includes a first set of sixty-four seventeen-bit registers,
8 the second register file includes a second set of sixty-four seventeen-bit
9 registers, the third register file includes a third set of sixty-four seventeen-bit
10 registers, and the fourth register file includes a fourth set of sixty-four
11 seventeen-bit registers.

1 30. (Cancelled)

- 1 31. (Previously Presented) The method of Claim 1, wherein the steps of determining the
- 2 intermediate result and determining the final result use an array of sixty-four
- 3 seventeen-bit by seventeen-bit modular multiplier circuits, and wherein the array of
- 4 sixty-four seventeen-bit by seventeen-bit modular multiplier circuits includes a
- 5 plurality of ratio four to two compressors that are organized into three levels and that
- 6 are executed in one clock cycle.
- 1 32. (Cancelled)
- 1 33. (Original) The method of Claim 1, wherein the steps of determining the intermediate
- 2 result and determining the final result use an array of sixty-four thirty-four-bit to
- 3 seventeen-bit modular reduction circuits that are executed in one clock cycle.
- 1 34. (Previously Presented) A computer-readable medium carrying one or more sequences
- 2 of instructions for encryption and decryption of electronic messages based on an
- 3 encryption protocol, which instructions, when executed by one or more processors,
- 4 cause the one or more processors to carry out the steps of:
- 5 receiving a first electronic message that is encrypted according to the encryption
- 6 protocol;
- 7 generating at least one part of a second electronic message, based on at least the first
- 8 electronic message, a modular operation that is based on two applications of
- 9 Montgomery's method, a first operand, a second operand, a modulus, and a
- 10 modular reduction using a negative multiplicative inverse of the modulus, and
- 11 wherein the instructions for generating the second electronic message further
- 12 comprise instructions which, when executed by one or more processors, cause
- 13 the one or more processors to carry out the steps of:
- 14 generating a first constant (R) based on the modulus (M) by selecting a second
- 15 constant (W) such that $W \geq 4M$ and determining the first constant (R)
- 16 according to the expression $R = W^2 \pmod{M}$;

17 in a first application of Montgomery's method, determining an intermediate
18 result based on at least Montgomery's method for the modular
19 operation, the first operand, and the first constant; and
20 in a second application of Montgomery's method, determining and storing in
21 memory a final result that comprises the at least one part of the second
22 electronic message, based on at least Montgomery's method for the
23 modular operation, the intermediate result, and the second operand.

1 35. (Currently Amended) An apparatus for encryption and decryption of electronic
2 messages based on an encryption protocol, comprising:
3 means for receiving a first electronic message that is encrypted according to the
4 encryption protocol;
5 means for generating at least one part of a second electronic message, based on at
6 least the first electronic message, a modular operation that is based on two
7 applications of Montgomery's method, a first operand, a second operand, a
8 modulus, and a modular reduction using a negative multiplicative inverse of
9 the modulus, and wherein the means for generating the second electronic
10 message further comprises:
11 means for generating a first constant (R) based on the modulus (M) by
12 selecting a second constant (W) such that $W \geq 4M$ and determining the
13 first constant (R) according to the expression $R = W^2 \pmod{M}$;[[;]]
14 means for determining, in a first application of Montgomery's method, an
15 intermediate result based on at least Montgomery's method for the
16 modular operation, the first operand, and the first constant; and
17 means for determining, in a second application of Montgomery's method, and
18 storing in memory a final result that comprises the at least one part of
19 the second electronic message, based on at least Montgomery's
20 method for the modular operation, the intermediate result, and the
21 second operand.

1 36. (Currently Amended) An apparatus for encryption and decryption of electronic
2 messages based on an encryption protocol, comprising:
3 an interface;
4 a processor coupled to the interface and receiving information from the interface; and
5 one or more stored sequences of instructions which, when executed by the processor,
6 cause the processor to carry out the steps of:
7 receiving a first electronic message that is encrypted according to the
8 encryption protocol;
9 generating at least one part of a second electronic message, based on at least
10 the first electronic message, a modular operation that is based on two
11 applications of Montgomery's method, a first operand, a second
12 operand, a modulus, and a modular reduction using a negative
13 multiplicative inverse of the modulus, and wherein the instructions for
14 generating the second electronic message further comprise instructions
15 which, when executed by the processors, cause the processor to carry
16 out the steps of:
17 generating a first constant (R) based on the modulus (M) by selecting a
18 second constant (W) such that $W \geq 4M$ and determining the
19 first constant (R) according to the expression $R = W^2$
20 $(\text{mod}(M));[[;]]$
21 in a first application of Montgomery's method, determining an
22 intermediate result based on at least Montgomery's method for
23 the modular operation, the first operand, and the first constant;
24 and
25 in a second application of Montgomery's method, determining and
26 storing in memory a final result that comprises the at least one
27 part of the second electronic message, based on at least
28 Montgomery's method for the modular operation, the
29 intermediate result, and the second operand.

1 37. (Cancelled)

1 38. (Previously Presented) The apparatus of Claim 36, wherein the second constant (W)
2 is not a power of two.

1 39. (Previously Presented) The apparatus of Claim 36, wherein the modular operation for
2 determining the intermediate result is a modular multiplication that is based on at
3 least the first operand, the first constant, a second constant, the modulus, and the
4 negative multiplicative inverse of the modulus, and wherein the instructions for
5 determining the intermediate result further comprise one or more stored sequences of
6 instructions which, when executed by the processor, cause the processor to carry out
7 the steps of:

8 determining a modified first operand based on the first operand and the first constant;
9 determining a modular reduction of the modified first operand based on the modified
10 first operand, the negative multiplicative inverse of the modulus, and the
11 second constant; and

12 determining the intermediate result based on the modified first operand, the modular
13 reduction of the modified first operand, the modulus, and the second constant.

1 40. (Previously Presented) The apparatus of Claim 36, wherein the modular operation for
2 determining the final result is a modular multiplication that is based on at least the
3 second operand, a second constant, the modulus, the negative multiplicative inverse
4 of the modulus, and the intermediate result, and wherein the instructions for
5 determining the final result further comprise one or more stored sequences of
6 instructions which, when executed by the processor, cause the processor to carry out
7 the steps of:

8 determining a modified second operand based on the second operand and the
9 intermediate result;

10 determining a modular reduction of the modified second operand based on the
11 modified second operand, the negative multiplicative inverse of the modulus,
12 and the second constant; and
13 determining the final result based on the modified second operand, the modular
14 reduction of the modified second operand, the modulus, and the second
15 constant.

1 41. (Previously Presented) The apparatus of Claim 36, wherein the modular operation for
2 determining the intermediate result is a modular exponentiation that is based on at
3 least the first operand, the first constant, a second constant, the modulus, and the
4 negative multiplicative inverse of the modulus, and wherein the instructions for
5 determining the intermediate result further comprise one or more stored sequences of
6 instructions which, when executed by the processor, cause the processor to carry out
7 the steps of:
8 determining a modified first operand based on the first operand and the intermediate
9 result;
10 determining a modular reduction of the modified second operand based on the
11 modified second operand, the negative multiplicative inverse of the modulus,
12 and the second constant; and
13 determining the intermediate result based on the modified second operand, the
14 modular reduction of the modified second operand, the modulus, and the
15 second constant.

1 42. (Currently Amended) The apparatus of Claim 36, wherein the modular operation for
2 determining the final result is a modular exponentiation that is based on at least the
3 second operand, a second constant, the modulus, a the negative multiplicative inverse
4 of the modulus, and the intermediate result, wherein the second operand includes a
5 plurality of digits, and wherein the instructions for determining the final result further
6 comprise one or more stored sequences of instructions which, when executed by the
7 processor, cause the processor to carry out the steps of:
8 specifying a previous final result as having a value of one and a previous intermediate
9 result as the intermediate result;
10 for each digit of the plurality of digits included in the second operand, carrying out the
11 steps of:
12 when each digit of the plurality of digits has the value of one, then carrying
13 out the steps of:
14 determining an updated final result based on the previous final result
15 and the previous intermediate result;
16 determining a modular reduction of the updated final result based on
17 the updated final result, the negative multiplicative inverse of
18 the modulus, and the second constant;
19 determining a revised updated final result based on the updated final
20 result, the modular reduction of the final result, the modulus,
21 and the second constant;
22 if all digits of the plurality of digits have not been evaluated,
23 specifying the revised updated final result as the previous final
24 result; and
25 if all digits of the plurality of digits have been evaluated, specifying the
26 revised updated final result as the final result;
27 determining an updated intermediate result based on the previous intermediate
28 result;

29 determining a modular reduction of the updated intermediate result based on
30 the updated intermediate result, the negative multiplicative inverse of
31 the modulus, and the second constant;
32 determining a revised updated intermediate result based on the updated
33 intermediate result, the modular reduction of the updated intermediate
34 result, the modulus, and the first constant; and
35 specifying the revised updated intermediate result as the previous intermediate
36 result.

1 43. (Previously Presented) The apparatus of Claim 36, further comprising one or more
2 stored sequences of instructions which, when executed by the processor, cause the
3 processor to carry out the steps of:
4 generating a plurality of residual number system (RNS) representations, wherein the
5 plurality of RNS representations includes at least one RNS representation for
6 each of the first operand, the modulus, and the first constant;
7 wherein the instructions for determining the intermediate result one or more stored
8 sequences of instructions which, when executed by the processor, cause the
9 processor to carry out the step of generating the intermediate result based on
10 Montgomery's method for the modular operation and the plurality of RNS
11 representations; and
12 wherein the instructions for determining the final result includes one or more stored
13 sequences of instructions which, when executed by the processor, cause the
14 processor to carry out the step of generating the final result based on
15 Montgomery's method for the modular operation and the plurality of RNS
16 representations.

1 44. (Previously Presented) The apparatus of Claim 43, wherein:
2 the plurality of RNS representations includes a first set of RNS representations in a
3 first RNS base and a second set of RNS representations in a second RNS base;
4 the first RNS base extends the second RNS base;

5 the first RNS base includes a first group of sixty-four residues and the second RNS
6 base includes a second group of sixty-four residues;
7 each residue in both the first group and second group is a seventeen-bit residue;
8 each residue in the first group is relatively prime with respect to all other residues in
9 the first group and each residue in the second group is relatively prime with
10 respect to all other residues in the second group;
11 each residue in both the first group and the second group are selected from a range
12 of 2^{16} to 2^{17} ; and
13 further comprising one or more stored sequences of instructions which, when
14 executed by the processor, cause the processor to carry out the steps of:
15 converting, in eight clock cycles, a first RNS representation of the first set of
16 RNS representations in the first RNS base to a second RNS
17 representation of the second set of RNS representations in the second
18 RNS base;
19 performing operations involving each residue in the first group in parallel; and
20 performing operations involving each residue in the second group in parallel.

1 45. (Previously Presented) The apparatus of Claim 36, wherein the modular operation is
2 a modular multiplication and the instructions for generating the second electronic
3 message further comprises one or more stored sequences of instructions which, when
4 executed by the processor, cause the processor to carry out the step of:
5 while determining the intermediate result and determining the final result, storing
6 results of intermediate computations in a first register file and a second
7 register file, wherein the first register file includes a first set of sixty-four
8 seventeen-bit registers and the second register file includes a second set of
9 sixty-four seventeen-bit registers.

- 1 46. (Previously Presented) The apparatus of Claim 36, wherein the modular operation is
- 2 a modular exponentiation and the instructions for generating the second electronic
- 3 message further comprises one or more stored sequences of instructions which, when
- 4 executed by the processor, cause the processor to carry out the step of:
- 5 while determining the intermediate result and determining the final result, storing
- 6 results of intermediate computations, in a first register file a second register
- 7 file, a third register file, and a fourth register file; and
- 8 wherein the first register file includes a first set of sixty-four seventeen-bit registers,
- 9 the second register file includes a second set of sixty-four seventeen-bit
- 10 registers, the third register file includes a third set of sixty-four seventeen-bit
- 11 registers, and the fourth register file includes a fourth set of sixty-four
- 12 seventeen-bit registers.
- 1 47. (Previously Presented) The apparatus of Claim 36, wherein the instructions for
- 2 determining the intermediate result and the instructions for determining the final
- 3 result make use of an array of sixty-four seventeen-bit by seventeen-bit modular
- 4 multiplier circuits, and wherein the array of sixty-four seventeen-bit by seventeen-bit
- 5 modular multiplier circuits includes a plurality of ratio four to two compressors that
- 6 are organized into three levels and that are executed in one clock cycle.
- 1 48. (Previously Presented) The apparatus of Claim 36, wherein the instructions for
- 2 determining the intermediate result and the instructions for determining the final
- 3 result make use of an array of sixty-four thirty-four-bit to seventeen-bit modular
- 4 reduction circuits that are executed in one clock cycle.
- 1 49. (Cancelled)
- 1 50. (Previously Presented) The computer-readable medium of Claim 34, wherein the
- 2 second constant (W) is not a power of two.

- 1 51. (Previously Presented) The computer-readable medium of Claim 34, wherein the
- 2 modular operation for determining the intermediate result is a modular multiplication
- 3 that is based on at least the first operand, the first constant, a second constant, the
- 4 modulus, and the negative multiplicative inverse of the modulus, and wherein the
- 5 instructions for determining the intermediate result further comprise one or more
- 6 sequences of instructions, which when executed by one or more processors, cause the
- 7 one or more processors to carry out the steps of:
- 8 determining a modified first operand based on the first operand and the first constant;
- 9 determining a modular reduction of the modified first operand based on the modified
- 10 first operand, the negative multiplicative inverse of the modulus, and the
- 11 second constant; and
- 12 determining the intermediate result based on the modified first operand, the modular
- 13 reduction of the modified first operand, the modulus, and the second constant.

- 1 52. (Previously Presented) The computer-readable medium of Claim 34, wherein the
- 2 modular operation for determining the final result is a modular multiplication that is
- 3 based on at least the second operand, a second constant, the modulus, the negative
- 4 multiplicative inverse of the modulus, and the intermediate result, and wherein the
- 5 instructions for determining the final result further comprise one or more sequences of
- 6 instructions, which when executed by one or more processors, cause the one or more
- 7 processors to carry out the steps of:
- 8 determining a modified second operand based on the second operand and the
- 9 intermediate result;
- 10 determining a modular reduction of the modified second operand based on the
- 11 modified second operand, the negative multiplicative inverse of the modulus,
- 12 and the second constant; and
- 13 determining the final result based on the modified second operand, the modular
- 14 reduction of the modified second operand, the modulus, and the second
- 15 constant.

1 53. (Previously Presented) The computer-readable medium of Claim 34, wherein the
2 modular operation for determining the intermediate result is a modular exponentiation
3 that is based on at least the first operand, the first constant, a second constant, the
4 modulus, and the negative multiplicative inverse of the modulus, and wherein the
5 instructions for determining the intermediate result further comprise one or more
6 sequences of instructions, which when executed by one or more processors, cause the
7 one or more processors to carry out the steps of:
8 determining a modified first operand based on the first operand and the intermediate
9 result;
10 determining a modular reduction of the modified second operand based on the
11 modified second operand, the negative multiplicative inverse of the modulus,
12 and the second constant; and
13 determining the intermediate result based on the modified second operand, the
14 modular reduction of the modified second operand, the modulus, and the
15 second constant.

1 54. (Previously Presented) The computer-readable medium of Claim 34, wherein the
2 modular operation for determining the final result is a modular exponentiation that is
3 based on at least the second operand, a second constant, the modulus, the negative
4 multiplicative inverse of the modulus, and the intermediate result, wherein the second
5 operand includes a plurality of digits, and wherein the instructions for determining the
6 final result further comprise one or more sequences of instructions, which when
7 executed by one or more processors, cause the one or more processors to carry out the
8 steps of:
9 specifying a previous final result as having a value of one and a previous intermediate
10 result as the intermediate result;
11 for each digit of the plurality of digits included in the second operand, carrying out the
12 steps of:
13 when each digit of the plurality of digits has the value of one, then carrying
14 out the steps of:

15 determining an updated final result based on the previous final result
16 and the previous intermediate result;
17 determining a modular reduction of the updated final result based on
18 the updated final result, the negative multiplicative inverse of
19 the modulus, and the second constant;
20 determining a revised updated final result based on the updated final
21 result, the modular reduction of the final result, the modulus,
22 and the second constant;
23 if all digits of the plurality of digits have not been evaluated,
24 specifying the revised updated final result as the previous final
25 result; and
26 if all digits of the plurality of digits have been evaluated, specifying the
27 revised updated final result as the final result;
28 determining an updated intermediate result based on the previous intermediate
29 result;
30 determining a modular reduction of the updated intermediate result based on
31 the updated intermediate result, the negative multiplicative inverse of
32 the modulus, and the second constant;
33 determining a revised updated intermediate result based on the updated
34 intermediate result, the modular reduction of the updated intermediate
35 result, the modulus, and the first constant; and
36 specifying the revised updated intermediate result as the previous intermediate
37 result.

- 1 55. (Previously Presented) The computer-readable medium of Claim 34, further
2 comprising one or more sequences of instructions, which when executed by one or
3 more processors, cause the one or more processors to carry out the step of:
4 generating a plurality of residual number system (RNS) representations, wherein the
5 plurality of RNS representations includes at least one RNS representation for
6 each of the first operand, the modulus, and the first constant;

7 wherein the instructions for determining the intermediate result includes one or more
8 sequences of instructions, which when executed by one or more processors,
9 cause the one or more processors to carry out the step of generating the
10 intermediate result based on Montgomery's method for the modular operation
11 and the plurality of RNS representations; and

12 wherein the instructions for determining the final result includes one or more
13 sequences of instructions, which when executed by one or more processors,
14 cause the one or more processors to carry out the step of generating the final
15 result based on Montgomery's method for the modular operation and the
16 plurality of RNS representations.

1 56. (Previously Presented) The computer-readable medium of Claim 55, wherein:
2 the plurality of RNS representations includes a first set of RNS representations in a
3 first RNS base and a second set of RNS representations in a second RNS base;
4 the first RNS base extends the second RNS base;
5 the first RNS base includes a first group of sixty-four residues and the second RNS
6 base includes a second group of sixty-four residues;
7 each residue in both the first group and second group is a seventeen-bit residue;
8 each residue in the first group is relatively prime with respect to all other residues in
9 the first group and each residue in the second group is relatively prime with
10 respect to all other residues in the second group;
11 each residue in both the first group and the second group are selected from a range
12 of 2^{16} to 2^{17} ; and
13 further comprising one or more sequences of instructions, which when executed by
14 one or more processors, cause the one or more processors to carry out the
15 steps of:
16 converting, in eight clock cycles, a first RNS representation of the first set of
17 RNS representations in the first RNS base to a second RNS
18 representation of the second set of RNS representations in the second
19 RNS base;

20 performing operations involving each residue in the first group in parallel; and
21 performing operations involving each residue in the second group in parallel.

1 57. (Previously Presented) The computer-readable medium of Claim 34, wherein the
2 modular operation is a modular multiplication and the instructions for generating the
3 second electronic message further comprises one or more sequences of instructions,
4 which when executed by one or more processors, cause the one or more processors to
5 carry out the step of:
6 while determining the intermediate result and determining the final result, storing
7 results of intermediate computations in a first register file and a second
8 register file, wherein the first register file includes a first set of sixty-four
9 seventeen-bit registers and the second register file includes a second set of
10 sixty-four seventeen-bit registers.

1 58. (Previously Presented) The computer-readable medium of Claim 34, wherein the
2 modular operation is a modular exponentiation and the instructions for generating the
3 second electronic message further comprises one or more sequences of instructions,
4 which when executed by one or more processors, cause the one or more processors to
5 carry out the step of:
6 while determining the intermediate result and determining the final result, storing
7 results of intermediate computations, in a first register file a second register
8 file, a third register file, and a fourth register file; and
9 wherein the first register file includes a first set of sixty-four seventeen-bit registers,
10 the second register file includes a second set of sixty-four seventeen-bit
11 registers, the third register file includes a third set of sixty-four seventeen-bit
12 registers, and the fourth register file includes a fourth set of sixty-four
13 seventeen-bit registers.

- 1 59. (Previously Presented) The computer-readable medium of Claim 34, wherein the
- 2 instructions for determining the intermediate result and the instructions for
- 3 determining the final result make use of an array of sixty-four seventeen-bit by
- 4 seventeen-bit modular multiplier circuits, and wherein the array of sixty-four
- 5 seventeen-bit by seventeen-bit modular multiplier circuits includes a plurality of ratio
- 6 four to two compressors that are organized into three levels and that are executed in
- 7 one clock cycle.
- 1 60. (Previously Presented) The computer-readable medium of Claim 34, wherein the
- 2 instructions for determining the intermediate result and the instructions for
- 3 determining the final result make use of an array of sixty-four thirty-four-bit to
- 4 seventeen-bit modular reduction circuits that are executed in one clock cycle.
- 1 61. (Cancelled)
- 1 62. (Previously Presented) The apparatus of Claim 35, wherein the second constant (W)
- 2 is not a power of two.
- 1 63. (Previously Presented) The apparatus of Claim 35, wherein the modular operation for
- 2 determining the intermediate result is a modular multiplication that is based on at
- 3 least the first operand, the first constant, a second constant, the modulus, and a the
- 4 negative multiplicative inverse of the modulus, and wherein the means for
- 5 determining the intermediate result further comprises:
6 means for determining a modified first operand based on the first operand and the first
- 7 constant;
- 8 means for determining a modular reduction of the modified first operand based on the
- 9 modified first operand, the negative multiplicative inverse of the modulus, and
- 10 the second constant; and

11 means for determining the intermediate result based on the modified first operand, the
12 modular reduction of the modified first operand, the modulus, and the second
13 constant.

1 64. (Previously Presented) The apparatus of Claim 35, wherein the modular operation for
2 determining the final result is a modular multiplication that is based on at least the
3 second operand, a second constant, the modulus, the negative multiplicative inverse
4 of the modulus, and the intermediate result, and wherein the means for determining
5 the final result further comprises:

6 means for determining a modified second operand based on the second operand and
7 the intermediate result;

8 means for determining a modular reduction of the modified second operand based on
9 the modified second operand, the negative multiplicative inverse of the
10 modulus, and the second constant; and

11 means for determining the final result based on the modified second operand, the
12 modular reduction of the modified second operand, the modulus, and the
13 second constant.

1 65. (Previously Presented) The apparatus of Claim 35, wherein the modular operation for
2 determining the intermediate result is a modular exponentiation that is based on at
3 least the first operand, the first constant, a second constant, the modulus, and the
4 negative multiplicative inverse of the modulus, and wherein the means for
5 determining the intermediate result further comprises:

6 means for determining a modified first operand based on the first operand and the
7 intermediate result;

8 means for determining a modular reduction of the modified second operand based on
9 the modified second operand, the negative multiplicative inverse of the
10 modulus, and the second constant; and

11 means for determining the intermediate result based on the modified second operand,
12 the modular reduction of the modified second operand, the modulus, and the
13 second constant.

1 66. (Previously Presented) The apparatus of Claim 35, wherein the modular operation for
2 determining the final result is a modular exponentiation that is based on at least the
3 second operand, a second constant, the modulus, the negative multiplicative inverse
4 of the modulus, and the intermediate result, wherein the second operand includes a
5 plurality of digits, and wherein the means for determining the final result further
6 comprises:

7 means for specifying a previous final result as having a value of one and a previous
8 intermediate result as the intermediate result;

9 means for performing, for each digit of the plurality of digits included in the second
10 operand, the following steps:

11 when each digit of the plurality of digits has the value of one, then performing
12 the computer-implemented steps of:

13 determining an updated final result based on the previous final result
14 and the previous intermediate result;

15 determining a modular reduction of the updated final result based on
16 the updated final result, the negative multiplicative inverse of
17 the modulus, and the second constant;

18 determining a revised updated final result based on the updated final
19 result, the modular reduction of the final result, the modulus,
20 and the second constant;

21 if all digits of the plurality of digits have not been evaluated,
22 specifying the revised updated final result as the previous final
23 result; and

24 if all digits of the plurality of digits have been evaluated, specifying the
25 revised updated final result as the final result;

26 determining an updated intermediate result based on the previous intermediate
27 result;

28 determining a modular reduction of the updated intermediate result based on
29 the updated intermediate result, the negative multiplicative inverse of
30 the modulus, and the second constant;

31 determining a revised updated intermediate result based on the updated
32 intermediate result, the modular reduction of the updated intermediate
33 result, the modulus, and the first constant; and
34 specifying the revised updated intermediate result as the previous intermediate
35 result.

1 67. (Previously Presented) The apparatus of Claim 35, further comprising:
2 means for generating a plurality of residual number system (RNS) representations,
3 wherein the plurality of RNS representations includes at least one RNS
4 representation for each of the first operand, the modulus, and the first
5 constant;
6 wherein the means for determining the intermediate result includes means for
7 generating the intermediate result based on Montgomery's method for the
8 modular operation and the plurality of RNS representations; and
9 wherein the means for determining the final result includes means for generating the
10 final result based on Montgomery's method for the modular operation and the
11 plurality of RNS representations.

1 68. (Previously Presented) The apparatus of Claim 67, wherein:
2 the plurality of RNS representations includes a first set of RNS representations in a
3 first RNS base and a second set of RNS representations in a second RNS base;
4 the first RNS base extends the second RNS base;
5 the first RNS base includes a first group of sixty-four residues and the second RNS
6 base includes a second group of sixty-four residues;
7 each residue in both the first group and second group is a seventeen-bit residue;
8 each residue in the first group is relatively prime with respect to all other residues in
9 the first group and each residue in the second group is relatively prime with
10 respect to all other residues in the second group;
11 each residue in both the first group and the second group are selected from a range
12 of 2^{16} to 2^{17} ; and

13 further comprising:

14 means for converting, in eight clock cycles, a first RNS representation of the
15 first set of RNS representations in the first RNS base to a second RNS
16 representation of the second set of RNS representations in the second
17 RNS base;

18 means for performing operations involving each residue in the first group in
19 parallel; and

20 means for performing operations involving each residue in the second group in
21 parallel.

1 69. (Previously Presented) The apparatus of Claim 35, wherein the modular operation is
2 a modular multiplication and the means for generating the second electronic message
3 further comprises:

4 means for storing, while determining the intermediate result and determining the final
5 result, results of intermediate computations in a first register file and a second
6 register file, wherein the first register file includes a first set of sixty-four
7 seventeen-bit registers and the second register file includes a second set of
8 sixty-four seventeen-bit registers.

1 70. (Previously Presented) The apparatus of Claim 35, wherein the modular operation is
2 a modular exponentiation and the means for generating the second electronic message
3 further comprises:

4 means for storing, while determining the intermediate result and determining the final
5 result, results of intermediate computations, in a first register file a second
6 register file, a third register file, and a fourth register file; and
7 wherein the first register file includes a first set of sixty-four seventeen-bit registers,
8 the second register file includes a second set of sixty-four seventeen-bit
9 registers, the third register file includes a third set of sixty-four seventeen-bit
10 registers, and the fourth register file includes a fourth set of sixty-four
11 seventeen-bit registers.

- 1 71. (Previously Presented) The apparatus of Claim 35, wherein the means for
- 2 determining the intermediate result and the means for determining the final result use
- 3 an array of sixty-four seventeen-bit by seventeen-bit modular multiplier circuits, and
- 4 wherein the array of sixty-four seventeen-bit by seventeen-bit modular multiplier
- 5 circuits includes a plurality of ratio four to two compressors that are organized into
- 6 three levels and that are executed in one clock cycle.

- 1 72. (Previously Presented) The apparatus of Claim 35, wherein the means for determining
- 2 the intermediate result and the means for determining the final result use an array of
- 3 sixty-four thirty-four-bit to seventeen-bit modular reduction circuits that are executed
- 4 in one clock cycle.